

- Altibase 7.1.0.4.0 Patch Notes

- New Features

- BUG-47805 SRID(Spatial Reference IDentifier) interface 지원
- BUG-47873 GEOMETRY 컬럼의 SRID 속성에 대해서 Replication 지원
- BUG-47816 ST_Transform 함수 지원
- BUG-47857 ST_MakePoint 함수 지원
- BUG-47883 ST_MakeLine 함수 지원
- BUG-47919 ST_MakeEnvelope 함수에서 SRID 인자를 추가합니다.
- BUG-47966 ST_PolygonFromText 함수 지원
- BUG-47974 sctTableSpaceMgr::mMutex 제거
- BUG-47963 altiComp에서 동일한 레코드들의 건수를 보여주는 기능 추가

- Fixed Bugs

- BUG-47786 Inverse Join으로 인해, subquery unnest 시 결과 오류가 발생할 수 있습니다.
- BUG-47840 단순 Partition Insert시 Execution memory 최적화
- BUG-47868 APRE에서 Anonymous block 을 사용할 때, 내부적으로 변환되는 내장 SQL 의 메모리 사이즈가 잘못 계산되고 있습니다.
- BUG-47884 prepare시에 통계정보를 구성할 때 인덱스와 통계정보가 매칭되지 않습니다.
- BUG-47889 altipasswd에서 암호 대소문자 구분이 필요합니다.
- BUG-47900 이중화 update XSN 시 에러가 발생했을때 로그에 잘못된 SN값이 기록되고 있습니다.
- BUG-47901 이중화 sendStop 함수에서 timeout 이 발생하여도 timeout 처리가 되지 않습니다.
- BUG-47910 통계 수집되지 않은 테이블의 unique constraint가 통계에 일부 포함됩니다.
- BUG-47917 외부 프로시저 생성구문에서 parsing 오류가 발생하는 경우가 있습니다.
- BUG-47918 PARTITION SPLIT 시 동일한 파티션 이름을 사용한 경우, 이중화 DDL Sync 에서 비정상 종료할 수 있습니다.
- BUG-47944 메타버전 8.8.1또는 8.7.1에서 하위버전으로 메타 다운그레이드 수행시 실패할 수 있습니다.
- BUG-47945 Disk Buffer의 CheckPoint List 오류에 대한 디버깅 코드 추가
- BUG-47948 DROP TABLESPACE memory_tablespace 와 V\$DATABASE 조회 중 동시성 문제로 인해 서버가 비정상 종료할 수 있습니다.
- BUG-47953 USE_DW_BUFFER enable 환경에서, DW_BUFFER 를 이용해서 깨진 페이지(Corrupt Page)를 복구할 때 비정상 종료 할 수 있습니다.
- BUG-47954 active-active상황에서 동시에 truncate 할 경우 sender가 stop될수 있습니다.
- BUG-47970 Disk Index 에서 inconsistent page 에 대해서 page latch가 풀리지 않는 경우가 있습니다.

- [BUG-47986](#) disk table에서 INNER JOIN 시 OR 절 predicate이 constant filter와 합쳐져서 인덱스를 타는 경우 결과 오류
- [Changes](#)
 - [Version Info](#)
 - [호환성](#)
 - [프로퍼티](#)
 - [성능 뷰](#)
 - [메타 테이블](#)

Altibase 7.1.0.4.0 Patch Notes

New Features

BUG-47805 SRID(Spatial Reference Identifier) interface 지원

- **module** : st-spatial
- **Category** : Functionality
- **재현 빈도** : Always
- **증상** : ESRI의 ArcGIS 연동을 위하여 SRID(Spatial Reference Identifier) Interface 를 지원합니다. SRID(Spatial Reference Identifier)는 공간 객체를 구분하기 위해 지정하는 식별자입니다. SRID는 GEOMETRY 컬럼에 적용할 수 있으며, 테이블에 INSERT하는 GEOMETRY 객체에 SRID를 지정할 수도 있습니다. 컬럼이나 객체의 SRID를 지정하지 않는 경우 기본값으로 0이 됩니다. 공간 정보를 가지는 테이블 생성구문에 SRID 속성이 추가되었습니다.

```
CREATE TABLE table_name (
    column_name GEOMETRY [(precision)] [(SRID srid)];
```

GEOMETRY 표현형식에 아래의 2가지 방법이 추가되었습니다.

- EWKT(Extended Well-Known Text) 형식: WKT 형식에 공간 객체를 표현하는 SRID(Spatial Reference Identifier) 정보가 추가된 형식
- EWKB(Extended Well-Known Binary) 형식: WKB 형식에 공간 객체를 표현하는 SRID(Spatial Reference Identifier) 정보가 추가된 형식

기본함수로 [SRID](#), [SETSRID](#), [ASEWKT](#), [ASEWKB](#) 함수가 추가되었고,

공간객체 함수로 [GEOMFROMEWKT](#), [GEOMFROMEWKB](#) 함수가 추가되었습니다.

자세한 내용은 [Spatial SQL Refernce 매뉴얼](#)을 참고하시기 바랍니다.

GEOMETRY 컬럼에 대한 제약사항이 추가되었으므로, [SRID제약사항](#)를 참고하시기 바랍니다.

주의사항

기존에 GEOMETRY_COLUMNS 와 SPATIAL_REF_SYS 테이블을 사용하는 고객은 아래의 변경사항이 있으므로, 패치전에 숙지해야 합니다.

- **GEOMETRY_COLUMNS** 테이블이 메타테이블로 변경되었고, 내용도 자동으로 설정됩니다.
 - 기존에 GEOMETRY_COLUMNS를 수동으로 설정하기 위해서 사용했던 ADDGEOMETRYCOLUMNS, DROPGEOMETRYCOLUMNS 프로시저는 더이상 사용할 수 없습니다. 자동으로 설정되기 때문에, 더이상 필요하지 않아서 삭제되었습니다.
- **SPATIAL_REF_SYS** 테이블이 메타테이블로 변경되었습니다.
 - 패치를 하면, 기존에 SPATIAL_REF_SYS 테이블에 있던 내용은 모두 사라집니다.
 - 업그레이드 및 마이그레이션시에, 해당 정보는 메타테이블에 INSERT를 수행할수 없기 때문에 Migration tool에서는 지원하지 않습니다.
 - 기존에는 SPATIAL_REF_SYS 테이블에 Spatial Reference System 메타 데이터를 등록하기 위해서는 DML을 이용했지만, 패치후에는 **ADD_SPATIAL_REF_SYS**, **DELETE_SPATIAL_REF_SYS** 프로시저를 이용해야 합니다.

업/다운그레이드 주의 사항

- 메타버전이 8.7.1 에서 8.8.1로 변경되어 , GEOMETRY 타입 컬럼이 있는 경우 메타 버전 8.7.1 이하(7.1.0.3.9 이하)로 다운그레이드를 할 수 없습니다.
- GEOMETRY 타입 컬럼이 없는 경우는 업/다운그레이드 이슈 없습니다.

• 재현 방법

- 재현 절차
- 수행 결과
- 예상 결과

• Workaround

• 변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-47873 GEOMETRY 컬럼의 SRID 속성에 대해서 Replication 지원

- **module** : rp
- **Category** : Functionality
- **재현 빈도** : Always

- **증상** : GEOMETRY에 SRID 속성을 지원함에 따라 이중화에서도 SRID 속성을 지원하도록 수정하였습니다.

GEOMETRY 컬럼을 사용하지 않는 고객의 경우는 패치 전후로 이슈가 없습니다.

GEOMETRY 컬럼을 사용하는 고객은 아래의 주의 사항을 확인하시기 바랍니다.

GEOMETRY 컬럼을 사용하는 경우의 이중화 주의 사항

- SRID를 지원하는 버전(7.1.0.4.0 이상)에서 SRID를 지원하지 않는 버전(7.1.0.3.9 이하)으로 이중화는 불가능
- SRID를 지원하지 않는 버전(7.1.0.3.9 이하)에서 SRID를 지원하는 버전(7.1.0.4.0 이상)으로 이중화는 가능

- **재현 방법**

- 재현 절차
- 수행 결과
- 예상 결과

- **Workaround**

- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code

BUG-47816 ST_Transform 함수 지원

- **module** : st-function
- **Category** : Functionality
- **재현 빈도** : Always
- **증상** : 입력된 geometry의 좌표계를 변환하는 함수인 [ST_Transform](#) 함수를 지원합니다.

```
GEOMETRY ST_Transform( GEOMETRY, INTEGER to_srid );
GEOMETRY ST_Transform( GEOMETRY, VARCHAR to_proj4text );
GEOMETRY ST_Transform( GEOMETRY, VARCHAR from_proj4text, VARCHAR to_proj4text );
GEOMETRY ST_Transform( GEOMETRY, VARCHAR from_proj4text, INTEGER to_srid );
```

주의

이 함수는 PROJ 라이브러리를 사용하는 함수로 altibase 서버에 내장되어 있습니다. (PROJ 버전: 4.9.3)

추가된 ST_transform 함수 사용시 PROJ 라이브러리가 사용하는 메모리는 알티베이스에서 관리되지 않습니다.

Intel 환경의 Linux 에서만 사용할수 있으며, 다른 환경에서 호출시에는 아래와 같이 해당 함수를 찾을 수 없다는 오류가 발생합니다.

[ERR-31129 : Procedure or function not found.]

PROJ4TEXT 문법은 PROJ 라이브러리 버전 4 형식만 지원합니다.

- 재현 방법
- 재현 절차
- 수행 결과
- 예상 결과
- Workaround
- 변경사항
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47857 ST_MakePoint 함수 지원

- **module** : st
- **Category** : Enhancement
- **재현 빈도** : Always
- **증상** : [ST_MakePoint](#) 함수가 추가되었습니다.

```
ST_MAKEPOINT( x, y )
```

- 재현 방법
 - 재현 절차
 - 수행 결과
 - 예상 결과
- Workaround
- 변경사항
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47883 ST_MakeLine 함수 지원

- **module** : st

- **Category** : Enhancement
- **재현 빈도** : Always
- **증상** : [ST_MakeLine](#) 함수가 추가되었습니다.

```
ST_MAKELINE( GEOMETRY1, GEOMETRY2 )
```

- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47919 ST_MakeEnvelope 함수에서 SRID 인자를 추가합니다.

- **module** : st-function
- **Category** : Functionality
- **재현 빈도** : Always
- **증상** : [ST_MakeEnvelope](#) 함수에서 SRID 인자를 추가합니다.

```
ST_MAKEENVELOPE( X1, Y1, X2, Y2[, SRID=0] )
```

- **재현 방법**
 - 재현 절차

```
SELECT ASEWKT( MAKEENVELOPE( 0, -1, 2, 3, 1000000 ) ) FROM DUAL;
```

- **수행 결과**
- **예상 결과**

```
ASEWKT( MAKEENVELOPE( 0, -1, 2, 3, 1000000
```

```
-----  
SRID=1000000;POLYGON((0 -1, 2 -1, 2 3, 0 3, 0 -1))
```

- **Workaround**
- **변경사항**
 - Performance view

- Property
- Compile Option
- Error Code

BUG-47966 ST_PolygonFromText 함수 지원

- **module** : st-function
- **Category** : Enhancement
- **재현 빈도** : Always
- **증상** : [ST_PolygonFromText](#) 함수가 추가되었습니다.
구문은 아래와 같습니다.

```
ST_PolygonFromText( wkt [, SRID] )
```

- **재현 방법**
 - **재현 절차**

```
create table t1 ( geom geometry );
insert into t1
values(
ST_PolygonFromText('POLYGON((-71.1776585052917 42.3902909739571,-71.1776820268866 42.39
-71.1776063012595 42.3903825660754,-71.1775826583081 42.3903033653531,-71.1776585052917
));
insert into t1
values(
ST_PolygonFromText('POLYGON((-71.1776585052917 42.3902909739571,-71.1776820268866 42.39
-71.1776063012595 42.3903825660754,-71.1775826583081 42.3903033653531,-71.1776585052917
));
select asewkt( the_geom ) from t1;
```

- **수행 결과**
- **예상 결과**

```
1 row inserted.
1 row inserted.
ASEWKT(THE_GEOM)
-----
SRID=0;POLYGON((-71.17766 42.39029, -71.17768 42.39037, -71.17761 42.39038, -71.17758 4
SRID=140;POLYGON((-71.17766 42.39029, -71.17768 42.39037, -71.17761 42.39038, -71.17758
2 rows selected.
```

- **Workaround**

```
ST_PolyFromTextSET_SRID + ST_PolyFromText
```

- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code

BUG-47974 sctTableSpaceMgr::mMutex 제거

- **module** : sm_resource
- **Category** : Enhancement
- **재현 빈도** : Frequence
- **증상** : MAX_OPEN_DATAFILE_COUNT 프로퍼티 보다 사용하는 file이 더 많을 때 발생하는 mutex 병목을 제거 합니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47963 altiComp에서 동일한 레코드들의 건수를 보여주는 기능 추가

- **module** : ux-audit(altiComp)
- **Category** : Enhancement
- **재현 빈도** : Always
- **증상** : altiComp에 동일한 값의 레코드 갯수를 보여주는 기능(EQ)을 추가했습니다.

```
[EQ_T->EQ_T]
Fetch Rec In Master: 3
Fetch Rec In Slave : 3
MOSX = DF, Count :      1
MXSO = DF, Count :      1
MOSO = DF, Count :      1
MOSO = EQ, Count :      1
```

- **재현 방법**
 - 재현 절차

- 수행 결과
- 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

Fixed Bugs

BUG-47786 Inverse Join으로 인해, subquery unnest 시 결과 오류가 발생할 수 있습니다.

- **module** : qp
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : 2개 이상의 Disk 테이블 JOIN 에서 Where조건에 Subquery가 여러 개 존재하고, OR predicate 으로 Subquery가 사용된 경우, SEMI INVERSE JOIN 이나 ANTI INVERSE JOIN으로 수행되면, 결과값 오류가 발생할 수 있습니다. 해당 경우에 SEMI JOIN 또는 ANTI JOIN에 대해서 INVERSE JOIN이 수행되지 않도록 수정하였습니다.
- **재현 방법**
 - 재현 절차

```
drop table TB1;
drop table TB2;
drop table TB3;
drop table TB4;
drop table TB5;
```

```
create table TB1(
CUST_CODE                VARCHAR(30)                NOT NULL,
SW_VER                   VARCHAR(50)                NOT NULL
) tablespace sys_tbs_disk_data;
```

```
alter table TB1 add primary key( CUST_CODE );
```

```
insert into TB1 values ( 'A', '111');
insert into TB1 values ( 'B', '111');
insert into TB1 values ( 'C', '111');
insert into TB1 values ( 'D', '111');
insert into TB1 values ( 'E', '111');
insert into TB1 values ( 'F', '111');
insert into TB1 values ( 'G', '111');
insert into TB1 values ( 'H', '111');
```

```
create table TB2 (
RB_SKU_CODE              VARCHAR(30)                NOT NULL,
DEL_YN                   VARCHAR(1)                 NOT NULL,
USE_YN                   VARCHAR(1)                 NOT NULL
) tablespace sys_tbs_disk_data;;
```

```
insert into TB2 values ( 'a', 'Y', 'N');
insert into TB2 values ( 'b', 'Y', 'N');
insert into TB2 values ( 'c', 'Y', 'N');
insert into TB2 values ( 'd', 'Y', 'N');
insert into TB2 values ( 'e', 'Y', 'N');
insert into TB2 values ( 'f', 'Y', 'N');
insert into TB2 values ( 'g', 'Y', 'N');
insert into TB2 values ( 'h', 'Y', 'N');
```

```
alter table TB2 add primary key( RB_SKU_CODE );
```

```
create table TB3 (
FCT_CODE                VARCHAR(30)                NOT NULL,
PLANT_CODE              VARCHAR(30)                NOT NULL,
STRG_ID                 VARCHAR(50)                NOT NULL,
RB_SKU_CODE             VARCHAR(30)                NOT NULL,
MIX_REASON_CODE        VARCHAR(30)                NOT NULL,
SW_VER                  VARCHAR(50)                NOT NULL,
DEL_YN                  VARCHAR(1)                 NOT NULL,
USE_YN                  VARCHAR(1)                 NOT NULL
) tablespace sys_tbs_disk_data;;
```

```
insert into TB3 select 'C310A', 'V341', 'SSS'||level, 'SM6G975UZKATMB', '111', 'Z', 'N
```

```

insert into TB3 select 'C310A', 'V341', 'SSS' || '133', 'SM6G975UZKATMB', '111', 'Z', 'N'
insert into TB3 select 'C310A', 'V341', 'SSS' || '134', 'SM6G975UZKATMB', '222', 'Z', 'N'
insert into TB3 select 'C310A', 'V340', 'SSS' || '11', 'SM6G975UZKATMB', '111', 'Z', 'N'
insert into TB3 select 'C310A', 'V340', 'SSS' || '12', 'SM6G975UZKATMB', '222', 'Z', 'N'
insert into TB3 select 'C310A', 'V340', 'SSS' || '13', 'SM6G975UZKATMB', '222', 'Z', 'N'
insert into TB3 select 'C310A', 'V340', 'SSS' || '14', 'SM6G975UZKATMB', '222', 'Z', 'N'
insert into TB3 select 'C310A', 'V340', 'SSS' || '15', 'SM6G975UZKATMB', '111', 'Z', 'N'
insert into TB3 select 'C310A', 'V340', 'SSS' || '16', 'SM6G975UZKATMB', '222', 'Z', 'N'
insert into TB3 select 'C310A', 'V340', 'SSS' || '17', 'SM6G975UZKATMB', '222', 'Z', 'N'
insert into TB3 select 'C310A', 'V340', 'SSS' || '18', 'SM6G975UZKATMB', '222', 'Z', 'N'

```

```

create table TB4 (
  COMM_CODE varchar(30 ),
  TYPE_CODE varchar(30 ),
  DEL_YN          VARCHAR(1)          NOT NULL,
  USE_YN          VARCHAR(1)          NOT NULL
) tablespace sys_tbs_disk_data;

```

```

insert into TB4 select 'SSS' || level, 'PICK_STRG', 'N', 'Y' from dual connect by level <

```

```

alter table TB4 add primary key( TYPE_CODE, COMM_CODE );

```

```

create table TB5(
  COMM_CODE varchar(30 ),
  CUST_CODE varchar(30 ),
  CUST_TYPE_CODE varchar(30),
  CUST_COMM_CODE varchar(30),
  COB_MULTI_MIX_REASON_YN varchar(30),
  DEL_YN          VARCHAR(1)          NOT NULL,
  USE_YN          VARCHAR(1)          NOT NULL,
  ATTR_C1_CODE varchar(30)
)tablespace sys_tbs_disk_data;

```

```

insert into TB5 values ( '111', '3433418X', 'COBCUST', '111', 'Y', 'N', 'Y', '973');
insert into TB5 values ( '222', '3433418X', 'MULTI_MIX_REASON', '111', 'Y', 'N', 'Y', ' '
insert into TB5 values ( '111', '3433418X', 'MULTI_MIX_REASON', '222', 'Y', 'N', 'Y', ' '

```

```

drop index idx1;
create index idx1 on TB5 ( CUST_CODE, CUST_TYPE_CODE);

```

```

SELECT A.DEL_YN, A.USE_YN
FROM TB3 A
      , (SELECT DECODE(COUNT(T2.CUST_COMM_CODE),0,'N','Y') COB_MULTI_MIX_REASON_YN
        FROM TB5 T1
          , TB5 T2
        WHERE T1.CUST_CODE = '3433418X'
          AND T1.CUST_TYPE_CODE = 'COBCUST'
          AND T2.CUST_CODE = '3433418X'
          AND T1.DEL_YN = 'N'
          AND T1.USE_YN = 'Y'
          AND T2.CUST_TYPE_CODE = 'MULTI_MIX_REASON'

```

```

        AND T2.ATTR_C1_CODE LIKE '%' || '973' || '%'
        AND T2.DEL_YN = 'N'
        AND T2.USE_YN = 'Y') G
WHERE 1 = 1
    AND A.FCT_CODE = 'C310A'
    AND A.STRG_ID IN (SELECT COMM_CODE FROM TB4 WHERE TYPE_CODE = 'PICK_ST
    AND A.RB_SKU_CODE = 'SM6G975UZZKATMB'
    AND ((G.COB_MULTI_MIX_REASON_YN = 'N' AND A.MIX_REASON_CODE = '000')
        OR
        (G.COB_MULTI_MIX_REASON_YN = 'Y'
            AND A.MIX_REASON_CODE IN (SELECT T2.CUST_COMM_CODE
                FROM TB5 T1
                , TB5 T2
            WHERE T1.CUST_CODE = '3433418X'
                AND T1.CUST_TYPE_CODE = 'COBCUST'
                AND T2.CUST_CODE = '3433418X'
                AND T1.DEL_YN = 'N'
                AND T1.USE_YN = 'Y'
                AND T2.CUST_TYPE_CODE = 'MULTI_MIX_REASON'
                AND T2.ATTR_C1_CODE LIKE '%' || '973' || '%'
                AND T2.DEL_YN = 'N'
                AND T2.USE_YN = 'Y'))))
    AND ( A.RB_SKU_CODE, A.SW_VER ) NOT IN (SELECT /*+*/ B.RB_SKU_CODE , A.SW_VER
        FROM TB1 A, TB2 B
        WHERE B.RB_SKU_CODE = 'x'
            AND A.CUST_CODE = 'x'
            AND B.DEL_YN = 'N' AND B.USE_YN = 'Y')

    AND A.DEL_YN = 'N' AND A.USE_YN = 'Y'
;

```

○ 수행 결과

```

DEL_YN  USE_YN
-----
N  Y
N  Y
N  Y
N  Y
N  Y
N  Y
6 rows selected.

```

○ 예상 결과

N Y

- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47840 단순 Partition Insert시 Execution memory 최적화

- **module** : qp
- **Category** : Efficiency
- **재현 빈도** : Always
- **증상** : Partition table에 단순 insert 수행에도 내부적으로 partition 갯수만큼 cursor를 할당하여, Execution Memory를 많이 사용하는 문제가 있습니다. 단순 insert 일경우 내부 커서를 이용하여 Execution memory 사용량을 줄이도록 수정합니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47868 APRE에서 Anonymous block 을 사용할 때, 내부적으로 변환되는 내장 SQL의 메모리 사이즈가 잘못 계산되고 있습니다.

- **module** : mm-apre
- **Category** : Functional Error
- **재현 빈도** : Rare
- **증상** : APRE에서 Anonymous block 을 사용할때, 내부적으로 변환할 내장 SQL의 메모리 사이즈가 잘못 계산되는 문제가 있어, 수정합니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
- Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47884 prepare시에 통계정보를 구성할 때 인덱스와 통계정보가 매칭되지 않습니다.

- **module** : qp-dml-pvo
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : 쿼리 prepare 시에 통계정보를 구성할때 인덱스와 인덱스 통계정보가 매칭되지 않는 문제를 수정합니다.
- **재현 방법**
 - **재현 절차**

```
drop table t1;
create table t1(i1 varchar(10), i2 varchar(20), i3 varchar(30));
create index t1_idx1 on t1(i2);
create index t1_idx2 on t1(i3);
insert into t1 ( select level, mod(level, 30), mod(level, 40) from dual connect by leve
alter table t1 add constraint t1_pk primary key (i1);
create index t1_idx3 on t1(i2, i3);
exec gather_database_stats();
alter system set trclog_explain_graph=1;
alter session set explain plan = on;
select * from t1;
var v1 bigint;
var v2 bigint;
var v3 bigint;
var v4 bigint;
var v5 bigint;
var v6 bigint;
var v7 bigint;
exec GET_INDEX_STATS( 'SYS', 'T1_PK', NULL, :v1,:v2,:v3,:v4,:v5,:v6,:v7);
print v3;
exec GET_INDEX_STATS( 'SYS', 'T1_IDX1', NULL, :v1,:v2,:v3,:v4,:v5,:v6,:v7);
print v3;
exec GET_INDEX_STATS( 'SYS', 'T1_IDX2', NULL, :v1,:v2,:v3,:v4,:v5,:v6,:v7);
print v3;
exec GET_INDEX_STATS( 'SYS', 'T1_IDX3', NULL, :v1,:v2,:v3,:v4,:v5,:v6,:v7);
print v3;
```

- **수행 결과**

```

|  |== Index Information ==
|  | T1_IDX3 : 120
|  | T1_IDX1 : 1000
|  | T1_PK : 40
|  | T1_IDX2 : 30
iSQL> exec GET_INDEX_STATS( 'SYS', 'T1_PK', NULL, :v1,:v2,:v3,:v4,:v5,:v6,:v7);
Execute success.
iSQL> print v3;
NAME                                TYPE                                VALUE
-----
V3                                BIGINT                                1000
iSQL> exec GET_INDEX_STATS( 'SYS', 'T1_IDX1', NULL, :v1,:v2,:v3,:v4,:v5,:v6,:v7);
Execute success.
iSQL> print v3;
NAME                                TYPE                                VALUE
-----
V3                                BIGINT                                30
iSQL> exec GET_INDEX_STATS( 'SYS', 'T1_IDX2', NULL, :v1,:v2,:v3,:v4,:v5,:v6,:v7);
Execute success.
iSQL> print v3;
NAME                                TYPE                                VALUE
-----
V3                                BIGINT                                40
iSQL> exec GET_INDEX_STATS( 'SYS', 'T1_IDX3', NULL, :v1,:v2,:v3,:v4,:v5,:v6,:v7);
Execute success.
iSQL> print v3;
NAME                                TYPE                                VALUE
-----
V3                                BIGINT                                120

```

○ 예상 결과


```

|  |== Index Information ==
|  | T1_IDX3 : 120
|  | T1_IDX1 : 30
|  | T1_PK : 1000
|  | T1_IDX2 : 40
iSQL> exec GET_INDEX_STATS( 'SYS', 'T1_PK', NULL, :v1,:v2,:v3,:v4,:v5,:v6,:v7);
Execute success.
iSQL> print v3;
NAME                                TYPE                                VALUE
-----
V3                                BIGINT                                1000
iSQL> exec GET_INDEX_STATS( 'SYS', 'T1_IDX1', NULL, :v1,:v2,:v3,:v4,:v5,:v6,:v7);
Execute success.
iSQL> print v3;
NAME                                TYPE                                VALUE
-----
V3                                BIGINT                                30
iSQL> exec GET_INDEX_STATS( 'SYS', 'T1_IDX2', NULL, :v1,:v2,:v3,:v4,:v5,:v6,:v7);
Execute success.
iSQL> print v3;
NAME                                TYPE                                VALUE
-----
V3                                BIGINT                                40
iSQL> exec GET_INDEX_STATS( 'SYS', 'T1_IDX3', NULL, :v1,:v2,:v3,:v4,:v5,:v6,:v7);
Execute success.
iSQL> print v3;
NAME                                TYPE                                VALUE
-----
V3                                BIGINT                                120

```

• Workaround

pk를 먼저 생성한 후 인덱스 생성

• 변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-47889 altipasswd에서 암호 대소문자 구분이 필요합니다.

- **module** : ux-isql
- **Category** : Functionality
- **재현 빈도** : Always

- **증상** : altibase.properties 의 CASE_SENSITIVE_PASSWORD 값에 따라, altipasswd가 암호 대소문자 구분하도록 수정합니다.

- **재현 방법**

- **재현 절차**

1. altibase.properties에서 CASE_SENSITIVE_PASSWORD = 1 로 변경
2. server restart
3. iSQL에서 sys user의 패스워드를 "Manager1"로 변경
alter user sys identified by "Manager1";
exit;
4. server stop
5. server . is 스크립트에서 암호 변경
6. altipasswd에서 새 암호로 변경
7. server start

- **수행 결과**

```
% server start
```

```
-----
Altibase Client Query utility.
Release Version 7.1.0.3.9
Copyright 2000, ALTIBASE Corporation or its subsidiaries.
All Rights Reserved.
-----
ISQL_CONNECTION = UNIX, SERVER = localhost
[ERR-9100F : Incorrect Password]
```

- **예상 결과**

```
% server start
```

```
-----
Altibase Client Query utility.
Release Version 7.1.0.4.0
Copyright 2000, ALTIBASE Corporation or its subsidiaries.
All Rights Reserved.
-----
ISQL_CONNECTION = UNIX, SERVER = localhost
[ERR-910FB : Connected to idle instance]
Connecting to the DB server... Connected.
...
--- STARTUP Process SUCCESS ---
Command executed successfully.
```

- **Workaround**

- **변경사항**

- Performance view
- Property
- Compile Option

- Error Code

BUG-47900 이중화 update XSN 시 에러가 발생했을때 로그에 잘못된 SN값이 기록되고 있습니다.

- **module** : rp
- **Category** : Other
- **재현 빈도** : Always
- **증상** : 이중화 update XSN 중 에러가 발생했을 때 로그에 잘못된 SN 값이 기록되고 있어서 수정하였습니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47901 이중화 sendStop 함수에서 timeout 이 발생하여도 timeout 처리가 되지 않습니다.

- **module** : rp
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : 이중화 sendStop 함수에서 timeout 이 발생해도 timeout 처리가 되지 않는 경우가 있어, 수정합니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47910 통계 수집되지 않은 테이블의 unique constraint가 통계에 일부 포함됩니다.

- **module** : ux-aexport
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : 통계정보가 수집되지 않은 테이블의 unique constraint가 ALL_EXE_STATS.sql 파일에 일부 포함되는 결함을 수정하였습니다.
- **재현 방법**
 - **재현 절차**

```
#####
# iSQL
#####
/* 초기화 */
EXEC DELETE_DATABASE_STATS();
drop table t1 cascade;
drop table t2 cascade;
drop table t3 cascade;
/* 통계 수집 대상 테이블 */
create table t1 (c1 integer primary key, c2 char(10));
create table t3 (c1 integer primary key, c2 char(10));
insert into t1 values (1, 't1');
insert into t3 values (3, 't3');
/* 통계 정보 수집 */
EXEC GATHER_DATABASE_STATS();
/* 통계 수집 되지 않은 테이블과 constraints */
create table t2 (c1 integer primary key, c2 char(10), c3 char(10));
alter table t2 add unique("C2");
alter table t2 add unique("C3");
#####
# shell
#####
aexport -s 127.0.0.1 -u sys -p manager
cat ALL_EXE_STATS.sql;
```

- **수행 결과**

```
$ cat ALL_EXE_STATS.sql;
...
EXEC SET_TABLE_STATS('SYS', 'T1', NULL, 1, 1, 16, 0.000000);
EXEC SET_COLUMN_STATS('SYS', 'T1', 'C1', NULL, 1, 0, 4, '1', '1');
EXEC SET_COLUMN_STATS('SYS', 'T1', 'C2', NULL, 1, 0, 12, 't1', 't1');
EXEC SET_PRIMARY_KEY_STATS('SYS', 'T1', 1, 1, 0, 0, 0, 1);
EXEC SET_UNIQUE_KEY_STATS('SYS', 'T2', 'C2', EXEC SET_UNIQUE_KEY_STATS('SYS', '
EXEC SET_COLUMN_STATS('SYS', 'T3', 'C1', NULL, 1, 0, 4, '3', '3');
EXEC SET_COLUMN_STATS('SYS', 'T3', 'C2', NULL, 1, 0, 12, 't3', 't3');
EXEC SET_PRIMARY_KEY_STATS('SYS', 'T3', 1, 1, 0, 0, 0, 1);
```

- 예상 결과

```
$ cat ALL_EXE_STATS.sql;
...
EXEC SET_COLUMN_STATS('SYS', 'T1', 'C1',NULL, 1, 0, 4, '1', '1');
EXEC SET_COLUMN_STATS('SYS', 'T1', 'C2',NULL, 1, 0, 12, 't1', 't1');
EXEC SET_TABLE_STATS('SYS', 'T3', NULL, 1, 1, 16, 0.000000);
EXEC SET_COLUMN_STATS('SYS', 'T3', 'C1',NULL, 1, 0, 4, '3', '3');
EXEC SET_COLUMN_STATS('SYS', 'T3', 'C2',NULL, 1, 0, 12, 't3', 't3');
```

- Workaround

- 변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-47917 외부 프로시저 생성구문에서 parsing 오류가 발생하는 경우가 있습니다.

- **module** : qp-psm-trigger-pvo
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : 외부 프로시저 생성구문 중 "AS LANGUAGE C" 에서 parsing 오류가 발생하여, 수정합니다.
- **재현 방법**
 - 재현 절차

```
Drop Library TestLib2;
Drop Procedure SimpleExternalProcedure;

CREATE OR REPLACE LIBRARY TestLib2 AS 'str_uppercase2.so';

CREATE OR REPLACE PROCEDURE SimpleExternalProcedure (a1 in varchar2, a2 out varchar2)
/
```

- 수행 결과

```
[ERR-31001 : SQL syntax error
0001 : CREATE OR REPLACE PROCEDURE SIMPLEEXTERNALPROCEDURE (A1 in VARCHAR2, A2 out VARC
]

```

- 예상 결과

```
Create success.
```

- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47918 PARTITION SPLIT 시 동일한 파티션 이름을 사용한 경우, 이중화 DDL Sync 에서 비정상 종료할 수 있습니다.

- **module** : rp
- **Category** : Fatal
- **재현 빈도** : Always
- **증상** : PARTITION SPLIT 시 동일한 partition이름을 사용한 경우, 내부적으로 새로운 partition ID를 갖게 되는데, 이중화에서 해당 정보를 업데이트 하지 않고, 이전의 정보를 참고하는 문제가 있어 비정상 종료 할 수 있습니다. PARTITION SPLIT을 할때 PARTITION ID가 변경되면 새로운 PARTITION ID를 사용하도록 수정하였습니다.

- **재현 방법**

- **재현 절차**

```
create table part_tbl
(c1 integer primary key,
c2 char(1200)
)
partition by range (c1)
(partition d1 values default tablespace USER_DATA_MEM) tablespace USER_DATA_MEM;

alter table part_tbl split partition D1 AT ( 5001 ) INTO ( PARTITION P_0_5000 TABLESPACE
```

- **수행 결과**

비정상 종료

- **예상 결과**

success

- **Workaround**

```
alter table part_tbl split partition D1 AT ( 5001 ) INTO ( PARTITION P_0_5000 TABLESPACE USI
```

- **변경사항**

- Performance view
- Property
- Compile Option

- Error Code

BUG-47944 메타버전 8.8.1또는 8.7.1에서 하위버전으로 메타 다운그레이드 수행시 실패할 수 있습니다.

- **module** : qp-meta
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : AIX 에서 Altibase 7.1 의 메타버전이 아래와 같은 경우, 메타 다운그레이드 수행시 실패합니다.
 - 메타버전 -> 다운그레이드 하려는 메타 버전
 - 8.8.1 -> 8.7.1 이하
 - 8.7.1 -> 8.6.1 이하

이로 인해, 7.1.0.1.8 부터 메타다운그레이드 이슈가 있었는데, 7.1.0.4.0부터는 정상수행됩니다.

- **재현 방법**
 - **재현 절차**

`meta downgrade`
 - **수행 결과**

% is

```
-----
Altibase Client Query utility.
Release Version 7.1.0.4.0
Copyright 2000, ALTIBASE Corporation or its subsidiaries.
All Rights Reserved.
-----
```

```
ISQL_CONNECTION = TCP, SERVER = localhost, PORT_NO = 3096
iSQL> SELECT META_MAJOR_VER, META_MINOR_VER, META_PATCH_VER,
      2  PREV_META_MAJOR_VER, PREV_META_MINOR_VER, PREV_META_PATCH_VER
      3  FROM SYSTEM_.SYS_DATABASE_;
META_MAJOR_VER META_MINOR_VER META_PATCH_VER PREV_META_MAJOR_VER PREV_META_MINOR_VER
```

```
-----
PREV_META_PATCH_VER
```

```
-----
8          8          1          8          7
```

```
1
```

```
1 row selected.
```

```
% quit
```

```
% server stop
```

```
-----
Altibase Client Query utility.
Release Version 7.1.0.4.0
Copyright 2000, ALTIBASE Corporation or its subsidiaries.
All Rights Reserved.
-----
```

```
ISQL_CONNECTION = UNIX, SERVER = localhost
```

```
Ok..Shutdown Proceeding....
```

```
TRANSITION TO PHASE : Shutdown Altibase
```

```
  [RP] Finalization : PASS
```

```
shutdown immediate success.
```

```
% server downgrade
```

```
-----
Altibase Client Query utility.
Release Version 7.1.0.4.0
Copyright 2000, ALTIBASE Corporation or its subsidiaries.
All Rights Reserved.
-----
```

```
ISQL_CONNECTION = UNIX, SERVER = localhost
```

```
[ERR-910FB : Connected to idle instance]
```

```
Connecting to the DB server.. Connected.
```

```
TRANSITION TO PHASE : PROCESS
```

```
TRANSITION TO PHASE : CONTROL
```

```
TRANSITION TO PHASE : META
```

```
  [SM] Recovery Phase - 1 : Preparing Database
```

```
                        : Dynamic Memory Version => Parallel Loading
```

```
  [SM] Recovery Phase - 2 : Loading Database
```

```
  [SM] Recovery Phase - 3 : Skipping Recovery & Starting Threads...
```

```
                        Refining Disk Table
```

```
  [SM] Refine Memory Table : .....
```

```
  [SM] Rebuilding Indices [Total Count:131] .....
```



```
[FAILURE] Value overflow
```

Startup Failed....

○ 예상 결과

- **Workaround**

- Performance view
- Property
- Compile Option
- Error Code

- **module** : sm-disk-resource

- **Category** : Fatal

- 재현 빈도 : Rare

- **증상** : 내부적으로 Checkpoint 연결함수에서 잘못된 BCB(dirty Buffer Control Block)에 대한 예외처리가 누락되어 있어, 디버깅 코드 추가하였습니다.

- 재현 방법

- 재현 절차

● 수행 결과

- **예상 결과**

- **Workaround**

- Performance view
- Property
- Compile Option
- Error Code

BUG-47948 DROP TABLESPACE memory_tablespace 와 V\$DATABASE 조회 중 동시성 문제로 인해 서버가 비정상 종료할 수 있습니다.

- **module** : sm-mem-resource
- **Category** : Fatal
- **재현 빈도** : Frequence
- **증상** : DROP TABLESPACE memory_tablespace 와 V\$MEMBASE 조회 사이에 동시성 문제가 있어서 수정합니다.
추가로, *VMEM_TABLESPACES*, *VNLS_PARAMETERS*, *V\$DB_FREEPAGELISTS* 의 조회시에도 동일한 문제가 있어서 함께 수정합니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47953 USE_DW_BUFFER enable 환경에서, DW_BUFFER 를 이용해서 깨진 페이지(Corrupt Page)를 복구할 때 비정상 종료 할 수 있습니다.

- **module** : sm-disk-recovery
- **Category** : Fatal
- **재현 빈도** : Always
- **증상** : 서버 비정상 종료 이후, DW_BUFFER를 이용하여 깨진 페이지(Corrupt Page)를 복구할 때, 내부적으로 DW에서 Logging을 시도하여 비정상 종료하는 문제가 발생합니다. 불필요한 logging을 시도하지 않도록 수정하여 해결합니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47954 active-active 상황에서 동시에 truncate 할 경우 sender가 stop될수 있습니다.

- **module** : rp
- **Category** : Functional Error
- **재현 빈도** : Frequence
- **증상** : 이중화 Active-Active 상황에서 동시에 여러개의 테이블을 truncate 할 경우, receiver 에서 동시성 문제로 lock timeout이 발생하여 Sender가 stop되고 재시작 되지 않는 문제가 있어 수정합니다.
- **재현 방법**
 - **재현 절차**

```
CREATE TABLE T1 ( I1 INTEGER PRIMARY KEY, I2 CHAR(20), I3 VARCHAR(20) )
PARTITION BY RANGE(I1)
(
    PARTITION P1 VALUES LESS THAN (10),
    PARTITION P2 VALUES LESS THAN (20),
    PARTITION P3 VALUES LESS THAN (30),
    PARTITION P4 VALUES LESS THAN (40),
    PARTITION def VALUES DEFAULT
);
...
CREATE REPLICATION REP1 WITH '123.123.123.123', 21300
FROM SYS.T1 PARTITION P1 TO SYS.T1 PARTITION P1,
FROM SYS.T2 PARTITION P1 TO SYS.T2 PARTITION P1,
FROM SYS.T3 PARTITION P1 TO SYS.T3 PARTITION P1,
FROM SYS.T4 PARTITION P1 TO SYS.T4 PARTITION P1;
...
ALTER REPLICATION REP1 START;
ALTER REPLICATION REP2 START;
ALTER REPLICATION REP3 START;
ALTER REPLICATION REP4 START;
TRUNCATE TABLE T1;
TRUNCATE TABLE T2;
TRUNCATE TABLE T3;
TRUNCATE TABLE T4;
```

- **수행 결과**

receiver lock timeout 발생으로 인해 sender stop 되고 재시작하지 않음

- **예상 결과**

sender가 재시작 되어야 함

- **Workaround**
- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code

BUG-47970 Disk Index 에서 inconsistent page 에 대해서 page latch가 풀리지 않는 경우가 있습니다.

- **module** : sm-disk-index
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : Disk Index 에서 inconsistent page 에 대해서 page latch가 풀리지 않는 경우가 있습니다.

mini transaction 의 null 예외처리가 누락된 것이 원인으로, null 예외처리 시 page latch가 풀리도록 수정합니다.

- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-47986 disk table에서 INNER JOIN 시 OR 절 predicate이 constant filter와 합쳐져서 인덱스를 타는 경우 결과 오류

- **module** : qp
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : disk table에서 inner join 시 OR 절 predicate이 constant filter와 합쳐져서 인덱스를 타는 경우 결과 오류 수정
- **재현 방법**
 - 재현 절차

```

drop table t1;
drop table t2;
create table t1 (
MODULE_CD                                VARCHAR(20) primary key,
MODULE_NM                                VARCHAR(200)
) tablespace sys_tbs_disk_data;
insert into t1 (MODULE_CD) values ('OS');
insert into t1 (MODULE_CD) values ('CR');
insert into t1 (MODULE_CD) values ('MP');
insert into t1 (MODULE_CD) values ('BM');
insert into t1 (MODULE_CD) values ('BP');
insert into t1 (MODULE_CD) values ('CI');
insert into t1 (MODULE_CD) values ('CO');

create table t2 (
MODULE_CD                                VARCHAR(20) ,
CLASS_CD                                VARCHAR(20)
)tablespace sys_tbs_disk_data;
insert into t2 (MODULE_CD) values ('OS');
insert into t2 (MODULE_CD) values ('CR');
insert into t2 (MODULE_CD) values ('MP');
insert into t2 (MODULE_CD) values ('BM');
insert into t2 (MODULE_CD) values ('BP');
insert into t2 (MODULE_CD) values ('CI');
insert into t2 (MODULE_CD) values ('CO');
insert into t2 (MODULE_CD) values ('FI');
insert into t2 (MODULE_CD) values ('HR');
insert into t2 (MODULE_CD) values ('OS');
insert into t2 (MODULE_CD) values ('CR');
insert into t2 (MODULE_CD) values ('MP');
insert into t2 (MODULE_CD) values ('BM');
insert into t2 (MODULE_CD) values ('BP');
insert into t2 (MODULE_CD) values ('CI');
insert into t2 (MODULE_CD) values ('CO');
insert into t2 (MODULE_CD) values ('FI');
insert into t2 (MODULE_CD) values ('HR');
insert into t2 (MODULE_CD) values ('OS');
insert into t2 (MODULE_CD) values ('CR');
insert into t2 (MODULE_CD) values ('MP');
insert into t2 (MODULE_CD) values ('BM');
insert into t2 (MODULE_CD) values ('BP');
insert into t2 (MODULE_CD) values ('CI');
insert into t2 (MODULE_CD) values ('CO');
insert into t2 (MODULE_CD) values ('FI');
insert into t2 (MODULE_CD) values ('HR');

var FORM varchar(80);
exec :FORM := NULL;
PREPARE
      SELECT /*+ USE_INDEX_NL(cdcn, cmi) */ count(*)
      FROM t1 CMI

```

```
INNER JOIN t2 CDCM ON CDCM.MODULE_CD = CMI.MODULE_CD
WHERE ( :FORM IS NULL OR :FORM = '' OR CMI.MODULE_CD = NULL);
```

수행 결과

```
COUNT(*)
-----
0
1 row selected.
```

예상 결과

```
COUNT(*)
-----
21
1 row selected.
```

Workaround

```
PREPARE
SELECT /*+ USE_HASH(t1, t2) */ count(*)
FROM t1 CMI
INNER JOIN t2 CDCM ON CDCM.MODULE_CD = CMI.MODULE_CD
WHERE ( :FORM IS NULL OR :FORM = '' OR CMI.MODULE_CD = NULL);
```

변경사항

- Performance view
- Property
- Compile Option
- Error Code

Changes

Version Info

altibase version	database binary version	meta version	cm protocol version	replication protocol version	sharding version
7.1.0.4.0	6.5.1	8.8.1	7.1.7	7.4.6	2.2.1

Altibase 7.1 패치 버전별 히스토리는 [Version_Histories](#) 에서 확인할 수 있다.

호환성

Database binary version

데이터베이스 바이너리 버전은 변경되지 않았다.

데이터베이스 바이너리 버전은 데이터베이스 이미지 파일과 로그파일의 호환성을 나타낸다. 이 버전이 다른 경우의 패치(업그레이드 포함)는 데이터베이스를 재구성해야 한다.

Meta Version

BUG-47805 SRID 인터페이스 지원으로 메타 버전이 8.7.1에서 8.8.1로 변경되었다.

- GEOMETRY 타입 컬럼이 없는 경우 이전과 동일하게 업패치, 다운패치를 진행할 수 있습니다.
- GEOMETRY 타입 컬럼이 있는 경우 업패치는 이슈가 없지만, 메타 버전 8.7.1 이하(7.1.0.3.9 이하)로 다운그레이드를 할 수 없습니다.

패치를 롤백하려는 경우, [메타다운그레이드](#)를 참고한다.

CM protocol Version

통신 프로토콜 버전은 변경되지 않았다.

Replication protocol Version

BUG-47873으로 인해, 이중화 프로토콜 버전이 7.4.5에서 7.4.6으로 변경됨에 따라, GEOMETRY 컬럼을 사용하는 고객의 경우 하위 버전과 이중화가 되지 않습니다.

또한 내부적으로 ALA 프로토콜이 변경되어, 7.1.0.4.0 JDBC Adapter 와 oraAdapter 에서 아래의 오류 (BUG-48015)가 발생합니다. 이 문제는 다음번 패치(7.1.0.4.1)에서 반영될 예정입니다.

BUG-48015 ALA protocol 변경으로 인해 jdbcAdapter, oraAdapter가 7.1.0.3.9 이하 버전과 연동되지 않습니다.

Sharding Version

샤딩 버전은 변경 되지 않았다.

알티베이스 샤딩 프로토콜 및 메타는 상위, 하위 호환성을 보장하지 않는다. 즉, 샤딩 버전이 다른 경우, 재구성해야 한다.

프로퍼티

추가/변경/삭제된 프로퍼티 없음

성능 뷰

추가/변경/삭제된 성능뷰 없음.

메타 테이블

추가된 테이블

- [GEOMETRY_COLUMNS](#)
- [SPATIAL_REF_SYS](#)

삭제된 테이블

- STO_COLUMNS_
- STO_DATUMS_
- STO_ELLIPSOIDS_
- STO_GEOCCS_
- STO_GEOGCS_
- STO_GEOMETRY_COLUMNS
- STO_PRIMEMS_
- STO_PROJCS_
- STO_PROJECTIONS_
- STO_SPATIAL_REF_SYS
- STO_SRS_
- STO_USER_COLUMNS_